UNIT I WEB ESSENTIALS

Internet Overview - Fundamental computer network concepts - Web Protocols - URL – Domain Name- Web Browsers and Web Servers- Working principle of a Website –Creating a Website – Fundamentals of Client-side and server-side scripting.

1.1 Internet Overview

Internet is a global network that connects billions of computers across the world with each other and to the World Wide Web. It uses standard internet protocol suite (TCP/IP) to connect billions of computer users worldwide. It is set up by using cables such as optical fibers and other wireless and networking technologies. At present, internet is the fastest mean of sending or exchanging information and data between computers across the world.


is believed that the internet was developed by "Defense Advanced Projects Agency" (DARPA) department of the United States. And, it was first connected in 1969.

Why is the Internet Called a Network?

Internet is called a network as it creates a network by connecting computers and servers across the world using routers, switches and telephone lines, and other communication devices and channels. So, it can be considered a global network of physical cables such as copper telephone wires, fiber optic cables, tv cables, etc. Furthermore, even wireless connections like 3G, 4G, or Wi-Fi make use of these cables to access the Internet.

Internet is different from the World Wide Web as the World Wide Web is a network of computers and servers created by connecting them through the internet. So, the internet is the backbone of the web as it provides the technical infrastructure to establish the WWW and acts as a medium to transmit information from one computer to another computer. It uses web browsers to display the information on the client, which it fetches from web servers.

How internet works
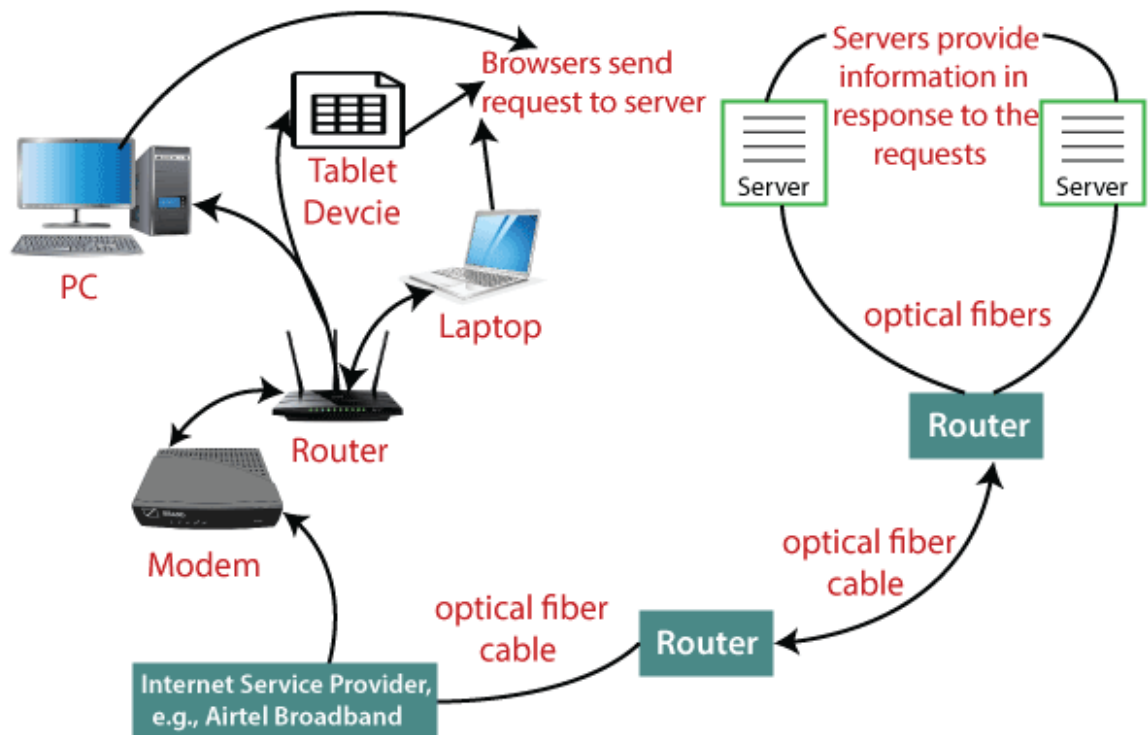
When you turn on your computer and type a domain name in the browser search bar, your browser sends a request to the DNS server to get the corresponding IP address. After getting the IP address, the browser forwards the request to the respective server.

Once the server gets the request to provide information about a particular website, the data starts flowing. The data is transferred through the optical fiber cables in digital format or in the form of light pulses. As the servers are placed at distant places, the data may have to travel thousands of miles through optical fiber cable to reach your computer.

The optical fiber is connected to a router, which converts the light signals into electrical signals. These electrical signals are transmitted to your laptop using an Ethernet cable. Thus, you receive the desired information through the internet, which is actually a cable that connects you with the server.

Furthermore, if you are using wireless internet using wifi or mobile data, the signals from the optical cable are first sent to a cell tower and from where it reaches to your cell phone in the form of electromagnetic waves.

1.2 Fundamentals of computer Network

"A Computer Network is defined as a set of two or more computers that are linked together?either via wired cables or wireless networks i.e., WiFi?with the purpose of communicating, exchanging, sharing or distributing data, files and resources."

Computer Networks are built using a collection of hardware (such as **routers, switches, hubs, and so forth)** and networking software (such as **operating systems, firewalls, or corporate applications).**

Basic Terminologies of Computer Networks

Network: A network is a collection of computers and devices that are connected together to enable communication and data exchange.

Nodes: Nodes are devices that are connected to a network. These can include computers, Servers, Printers, Routers, Switches, and other devices.

Protocol: A protocol is a set of rules and standards that govern how data is transmitted over a network. Examples of protocols include TCP/IP, HTTP, and FTP.

Topology: Network topology refers to the physical and logical arrangement of nodes on a network. The common network topologies include bus, star, ring, mesh, and tree.

Service Provider Networks: These types of Networks give permission to take Network Capacity and Functionality on lease from the Provider. Service Provider Networks include Wireless Communications, Data Carriers, etc.

IP Address: An IP address is a unique numerical identifier that is assigned to every device on a network. IP addresses are used to identify devices and enable communication between them.

DNS: The Domain Name System (DNS) is a protocol that is used to translate human-readable domain names (such as www.google.com) into IP addresses that computers can understand.

Firewall: A firewall is a security device that is used to monitor and control incoming and outgoing network traffic. Firewalls are used to protect networks from unauthorized access and other security threats.

Types of Enterprise Computer Networks

LAN: A Local Area Network (LAN) is a network that covers a small area, such as an office or a home. LANs are typically used to connect computers and other devices within a building or a campus.

WAN: A Wide Area Network (WAN) is a network that covers a large geographic area, such as a city, country, or even the entire world. WANs are used to connect LANs together and are typically used for long-distance communication.

Cloud Networks: Cloud Networks can be visualized with a Wide Area Network (WAN) as they can be hosted on public or private cloud service providers and cloud networks are available if there is a demand. Cloud Networks consist of Virtual Routers, Firewalls, etc.

1.3 WEB PROTOCOLS :

The Internet set of networks are all based on IP, the Internet Protocol. ``The Internet Protocol (IP) takes care of addressing, or making sure that the routers know what to do with your data when it arrives. Data is transmitted in a series of small chunks, called *packets*, each approximately 1200 characters in length.

- ➢ Communication protocol: how computers talk
  - o Cf. telephone "protocol": how you answer and end call, what language you speak, etc.Internet protocols developed as part of ARPANET research
  - o ARPANET began using TCP/IP in 1982
- ➢ Designed for use both within local area networks (LAN's) and between networks **IP** (**Internet Protocol**)**:**

Internet Protocol (IP) is the fundamental protocol defining the Internet (as the name implies!)

- ◆ **IP address:**
  - ✓ 32-bit number (in IPv4)
  - ✓ Associated with at most one device at a time (although device may have more than one)
  - ✓ Written as four dot-separated bytes, e.g. 192.0.34.166

Function of IP :

- ◆ Transfer data from source device to destination device
- ◆ IP source software creates a packet representing the data
  - ➢ Header: It consist of source and destination IP addresses, length of data, etc. and
  - ➢ Data itself
- ◆ If destination is on another LAN, packet is sent to a gateway that connects to more than one network
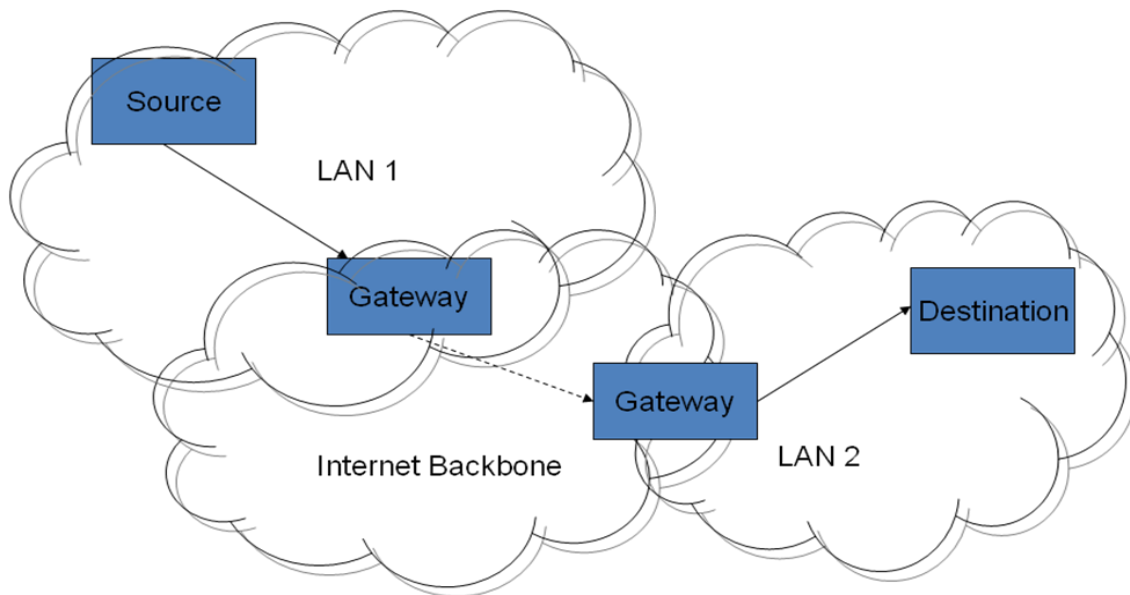
Figure show the Routing of IP

IP Address Classes :

xxx.xxx.xxx.xxx

Class A : 0.0.0.0 to 126.255.255.255   **Class B:** 128.0.0.0 to 191.255.255.255

Class C : 192.0.0.0 to 223.255.255.255   **Class D :** 224.0.0.0 to 239.255.255.255

Class E : 240.0.0.0 to 255.255.255.255

Class	First Octet Range	Max Hosts	Format
A	1-126	16M	NETID (0) — 1 Octet / HOSTID — 3 Octets
B	128-191	64K	NETID (1 0) — 2 Octets / HOSTID — 2 Octets
C	192-223	254	NETID (1 1 0) — 3 Octets / HOSTID — 1 Octet
D	224-239	N/A	Multicast Address (1 1 1 0)
E	240-255	N/A	Experimental (1 1 1 1)

Note :

Finally the address **127.0.0.1** is called the **loopback address** because it indicates the **localhost**.

Limitations of IP:

> ➢ No guarantee of packet delivery (packets can be dropped)
> ➢ Communication is one-way (source to destination)

**Transmission Control Protocol (TCP) :**

TCP (Transmission Control Protocol) is a set of rules (protocol) used along with the Internet Protocol (IP) to send data in the form of message units between computers over the Internet. While IP takes care of handling the actual delivery of the data, TCP takes care of keeping track of the individual units of data (called packets) that a message is divided into for efficient routing through the Internet.

- TCP adds concept of a connection on top of IP
  - ✓ Provides guarantee that packets delivered
  - ✓ Provide two-way (full duplex) communication
  - ✓ TCP rearranges data packets in the order specified.

Figure shows the TCP connection

- TCP also adds concept of a port
  - o TCP header contains port number representing an application program on the destination computer
  - o Some port numbers have standard meanings
    - Example: port 25 is normally used for email transmitted using the Simple Mail Transfer Protocol (SMTP)
  - o Other port numbers are available first-come-first served to any application
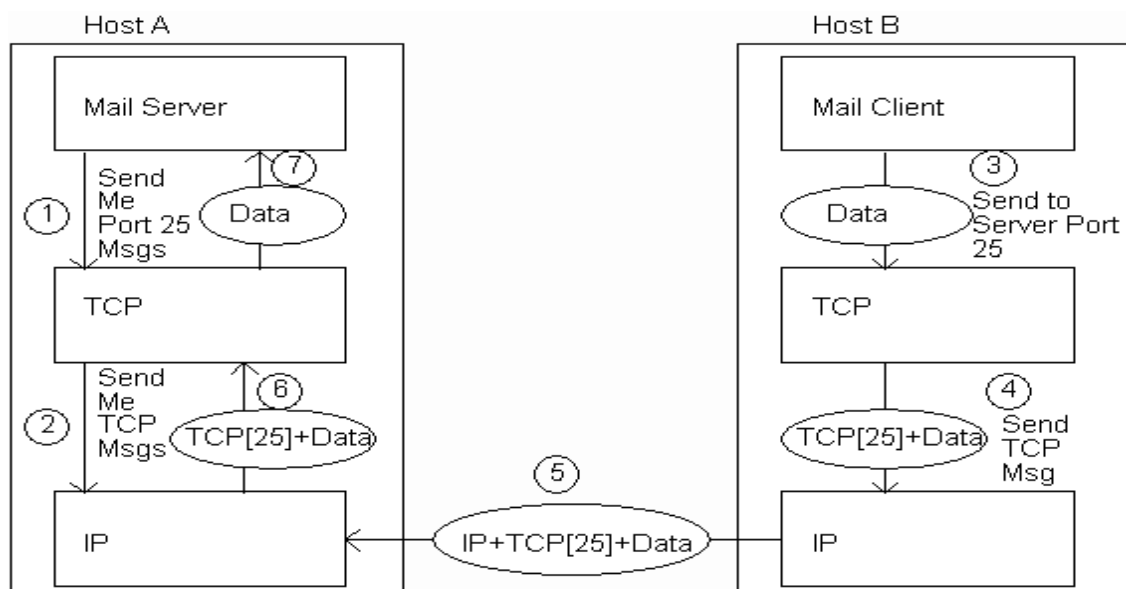
For TCP examples , HTTP, HTTPs, FTP, SMTP Telnet etc...

User Datagram Protocol (UDP) :

UDP (User Datagram Protocol) is a communications protocol that offers a limited amount of service when messages are exchanged between computers in a network that uses the

Internet Protocol . UDP provides two services not provided by the IP layer. It provides **port numbers** to help distinguish different user requests and, optionally, a **checksum** capability to verify that the data arrived intact.

- ◆ Like TCP in that:
    - ■ Builds on IP
    - ■ The speed for TCP in comparison with UDP is slower
    - ■ Provides port concept
- ◆ Unlike TCP in that:
    - ■ No connection concept
    - ■ No transmission guarantee
- ◆ Advantage of UDP vs. TCP:
    - ■ Lightweight, so faster for one-time messages
    - ■ UDP is faster because there is no error-checking for packets.
    - ■ UDP does not order packets. If ordering is required, it has to be managed by the application layer.

For UDP examples , DNS, DHCP, TFTP, SNMP, RIP, VOIP etc...

1. 4 URL: Uniform Resource Locator

URL stands for Uniform Resource Locator. It is the address of a resource, which can be a specific webpage or a file, on the internet. It is also *known as web address* when it is used with http. It was created in 1994 by Tim Berners-Lee. URL is a specific character string that is used to access data from the World Wide Web. It is a type of URI (Uniform Resource Identifier).

Every URL contains the following information:

- o The scheme name or protocol.
- o A colon, two slashes.
- o A host, normally called a domain name but sometimes as a literal IP address.
- o A colon followed by a port number.
- o Full path of the resource.

The URL of a web page is displayed above on the page in the address bar. A typical URL looks like this:

http://www.javatpoint.com/full-form

The above URL contains:

- o **protocol**: http
- o **host or domain**: www.javatpoint.com

      o **Path of the resource**: /full-form

A URL can be entered manually by typing it in the address bar of your web browser. If the URL does not contain a valid server, a browser may display a "Server not found" error and if the path in the URL is incorrect, the browser may display a "404 error". A URL does not contain spaces and uses forward slashes to represent different directories. So, dashes and underscores are used separate the words of a web address.

URI

URI stands for Uniform Resource Identifier. It is a generic term for all the name and addresses which show objects on the World Wide Web. It is generally a sequence of characters which identifies a logical resource or the name and location of a file or resource in a uniform format.

A URI can be of two types: Uniform Resource Locator (URL) and Uniform Resource Names (URNs). It enables resources to be accessed by other computers across a network or over the World Wide Web.

1.5 Domain Name Service ([DNS](#)) :

DNS is a look up mechanism for translating objects into other objects. It is a globally distributed , loosely coherent , scalable , reliable , dynamic database . It comprised of three components, A name space, Servers making that name space available, and Resolver (Clients) which query the servers about the name space.

✓ Data is maintained locally, but retrievable globally.

✓ The database is always internally consistent.

✓ No limit to the size of the database, number of queries.

✓ Data is replicated.

✓ Database can be updated dynamically.

✓ Maps names to values – resource records.

◆ DNS is the "phone book" for the Internet

➢ Map between host names and IP addresses

➢ DNS often uses UDP for communication

Host names

      o Labels separated by dots, e.g., www.example.org

      o Final label is *[top-level domain](#)*

          ▪ Generic: .com, .org, etc.

          ▪ Country-code: .us, .il, etc.

- Domains are divided into second-level domains, which can be further divided into subdomains, etc.

  E.g., in www.example.com, example is a second-level domain

- A host name plus domain name information is called the **fully qualified domain name** of the computer

  Above, www is the host name, www.example.com is the FQDN

- nslookup program provides command-line access to DNS (on most systems)

- looking up a host name given an IP address is known as a reverse lookup

  ➢ Recall that single host may have multiple IP addresses.

  ➢ Address returned is the canonical IP address specified in the DNS system.

- ipconfig (on windows) can be used to find the IP address (addresses) of your machine

ipconfig /displaydns displays the contents of the DNS Resolver Cache.

1.6 Web Browsers and Web Server

Web Browser

- Standard features
  - Save web page to disk
  - Find string in page
  - Fill forms automatically (passwords, CC numbers, …)
  - Set preferences (language, character set, cache and HTTP parameters)
  - Modify display style (e.g., increase font sizes)
  - Display raw HTML and HTTP header info (e.g., Last-Modified)
  - Choose browser themes (skins)
  - View history of web addresses visited
  - Bookmark favorite pages for easy return

- Additional functionality:
  - Execution of scripts (e.g., drop-down menus)
  - Event handling (e.g., mouse clicks)
  - GUI for controls (e.g., buttons)
  - Secure communication with servers
  - Display of non-HTML documents (e.g., PDF) via plug-ins
  -

Web Servers :

The primary feature of every web server is to accept HTTP requests from web clients and return an appropriate resource in the HTTP response.

A Web server is a computer program that delivers (serves) content, such as Web pages, using the Hypertext Transfer Protocol (HTTP), over the World Wide Web.

For example,

Microsoft's Internet Information Server (IIS) provides essentially all of the features found in Apache, although IIS have the drawback of running only on Windows systems, while Apache runs on Windows, Linux, and Macintosh systems.

- ◆ **Basic functionality:**
    - o Receive HTTP request via TCP
    - o Map Host header to specific virtual host (one of many host names sharing an IP address)
    - o Map Request-URI to specific resource associated with the virtual host
        - ▪ File: Return file in HTTP response
        - ▪ Program: Run program and return output in HTTP response
    - o Map type of resource to appropriate MIME type and use to set Content-Type header in HTTP response
    - o Log information about the request and response
- ➢ httpd: UIUC, primary Web server c. 1995
- ➢ Apache: "A patchy" version of httpd, now the most popular server (esp. on Linux platforms)
- ➢ IIS: Microsoft Internet Information Server
- ➢ **Tomcat Web Server**:
    - o Java-based
    - o Provides container (Catalina) for running Java servlets (HTML-generating programs) as back-end to Apache or IIS
    - o Can run stand-alone using Coyote HTTP front-end
- ➢ Some Coyote communication parameters:
    - o Allowed/blocked IP addresses
    - o Max. simultaneous active TCP connections
    - o Max. queued TCP connection requests
    - o "Keep-alive" time for inactive TCP connections
- ➢ Modify parameters to tune server performance

➢ Some Catalina container parameters:
  o Virtual host names and associated ports
  o Logging preferences
  o Mapping from Request-URI's to server resources
  o Password protection of resources
  o Use of server-side caching

1.7 CREATING WEB SITE

Web site is a **collection of web pages**. Hence for a website design we need to design the webpages.

Each webpage may contain texts, photos, videos, and social media buttons and so on. Technically, a webpage is a special type of document written in scripting languages such as HTML,CSS, JavaScript, PHP and so on. Web pages are written for web browsers.

The web browsers are the programs like **Internet Explorer, Google Chrome and Safari**.

Definition of website

Website is a collection of webpages that are grouped together to achieve certain task undersingle domain name.

Why do people visit website?

The most important reason is to **find the required information**. This could be anything from a student looking for images for a school project, to finding the latest stock quotes, for getting the address of the nearest restaurant and so on.

1. 8 Steps for creating the Website:

Step 1: Website creation:

Create a webpage using suitable **scripting language**. If any image is associated with this web page then convert this image into appropriate format (JPEG or GIF is preferable). Embed this image appropriately in this webpage.

Step 2: Choosing the web hosting services

• Web hosting company hosts your webpages on web server. Thus your website will be available to anyone who knows your **URL**. Most web hosting companies offer hosting services for both personal and business use. The web host provides you with Internet access, email accounts and space for a personal or business website.

• If you are building a website for business use, your webhost can register a personalized domain name for your website. If you are building a website for business use,

11

your web host can register a personalized domain name for your website.

• Small web sites (around 15-20 pages of contents) do not need much more than 1 or 2 MB of server space that hold all the HTML pages and graphics. Your web hosting package should provide atleast MB of space so your web page has room to grow.

Step 3: Registering Domain Name

Domain name is an **alias that points to actual location of your web site on web server**. Domain names are managed by the **Internet Corporation for Assigned Names andNumbers** (ICANN).

ICANN has agreements with a number of vendors to provide domain name registration services.

Step 4: Planning your website

• **Type:** The type of site you need. Is this a news or informational site, a site for a company or service, a non-profit or cause driven site, an E-commerce shop etc. Each of these kinds of site has a slightly different focus that will influence its design.

• **Navigation:** Navigation means indication that how users will move around your site affects itsinformation architecture as well as the overall usability of that site. Plan out the pages a site, create a sitemap, and develop a navigational structure from there.

• **Content:** The quality of your site's content will play an important role in its success. Content is everything that your pages will contain, such as text, images, video and more. Before you start designing or building pages, you should have a clear strategyfor the content that those pages will contain.

Step 5: Uploading Files

• To publish a website on the web, you must send the web pages created by you on the web server using **File Transfer Protocol (FTP).** Using some software such as Microsoft Visual Studio or Adobe Dreamweaver one can upload the files on the webserver.

Testing the website:

Testing must be performed throughout the development of website

• **Multiple Browsers:** It is necessary to display the website on as many web browsers as possible to ensure that the contents of the website are consistently displayed and the work doneis portable.

• **Multiple Operating Systems:** It is necessary to display the website on different operating systems.

• **Connection Speed:** Do not rely on the same connection speed when testing your

website, especially if you work in a corporate environment where the connection to the Internet usually is faster than the average user's. Also test the download time for different connection speed.

• **Device Types:** Test the websites on the computer's having different screen size. It is necessary to ensure that pages are displayed consistently on all screen size.

• **Links:** Use a link validation tool to ensure that all of your links connect to a live page.

• **Security Testing:** This step is necessary to test the security vulnerabilities in application running on the website. Security is an important part of any web development plan

1.9 WORKING PRINCIPLE OF A WEBSITE

**Step1:** User is requesting a webpage "www.google.com" in the web browser

Step2: Web Browser communicates with the DNS server for Domain name resolution.

Step 3: The DNS Server will look up the ip address for the corresponding webpage and send the IP address to web browser

Step 4: Web Browser receives the IP address for the given web page and then request the IP address to the Web server

Step 5: The web server searches for the IP address and sends the corresponding HTML and other files to the web browser

Step 6 : Web browser then displays the HTML and the other pages to the user
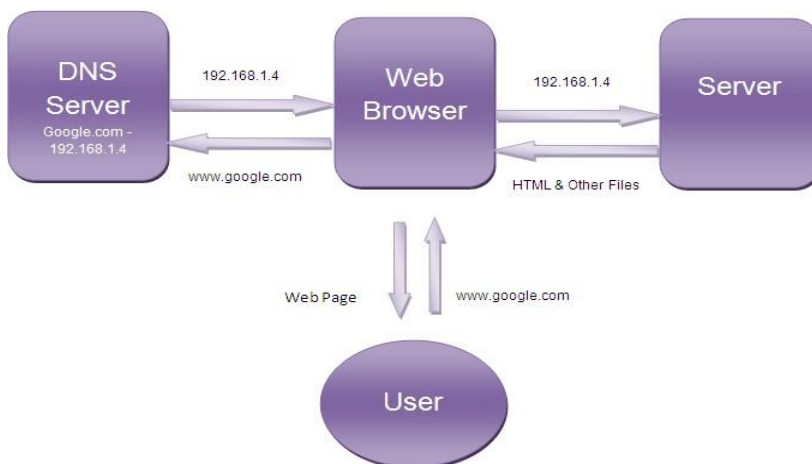
1.10 FUNDAMENTALS OF CLIENT AND SERVER SIDE SCRIPTING LANGUAGES

Definition of Scripting Language: A scripting language is a programming language designed for integrating and communicating with other programming languages.

✓ It the world's most popular programming language are HTML, JavaScript, VBScript, PHP, Perl, Python, Ruby, and ASP and so on.

13

- ✓ Scripting Languages are often interpreted (rather than compiled).
- ✓ The scripting languages are useful for producing dynamic web contents. That means web page can be changed using user input.
- ✓ Scripting languages are easy to learn.
- ✓ It requires minimum programming knowledge or experience to develop the web pages using scripting languages.
- ✓ The scripting languages allow simple creation and editing in variety of text editors.
- ✓ Using scripting languages we can develop dynamic and interactive web pages.
- ✓ There are some scripting languages that validate the information entered by the user.

Why Study JavaScript?

JavaScript is one of the **3 languages** all web developers **must** learn:

1. **HTML** to define the content of web pages
2. **CSS** to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages

TYPES OF SCRIPTING LANGUAGES

There are two types of scripting languages

- ✓ **Client side scripting language**
- ✓ **Server side scripting language**

Client Side Scripting Language:

The client side scripting is used to create the web pages as a request or response to server. These pages are displayed to the user on web browser.

For example: HTML, CSS, JavaScript, PHP.

Server Side Scripting Language:

Server side scripting is used to create web pages that provide some services. These scripts generally run on web servers.

For example: ASP, JSP, Servlet, PHP.

Difference between Client side and Server side scripting languages

Client-side

- Does not need interaction with the server
- Runs on the user's computer
- Reduces load on the server's proccessing unit
- Languages used: HTML, CSS, JavaScript

Server-side

- Requires interaction with the server
- Runs on the web server
- Allows the server to provide dynamic websites tailored to the user. Increases the processing load on server.
- Languages used: PHP, ASP.net, Python

JavaScript Introduction

✓ To enhance the functionality and appearance of Webpages.

✓ It is *De Facto* standard client-side scripting language for web-based applications due to its highly portable Nature.

Two purpose of JavaScript :

➢ It Introduces client-side scripting, which makes web pages more dynamic and interactive.

➢ It provides the programming foundation for more complex server side scripting.

➢ **Problem**: the JavaScript language itself has no input/output statements(!)

➢ **Solution**: Most browsers provide *de facto* standard I/O methods

▪ **alert**: pops up alert box containing text

▪ **prompt:** pops up window where user can enter text

JavaScript Basic Syntax :

JavaScript is a scripting language mainly used for writing dynamic Web pages. When a script written in JavaScript is embedded in a Web page, it will be executed by the Web browser on the client machine.

There are **three general areas that Javascript can be placed** for use in a web page.

1. Inside the **head** tag (i.e. a script run on some event)
2. Within the **body** tag (i.e. script to run when the page loads)
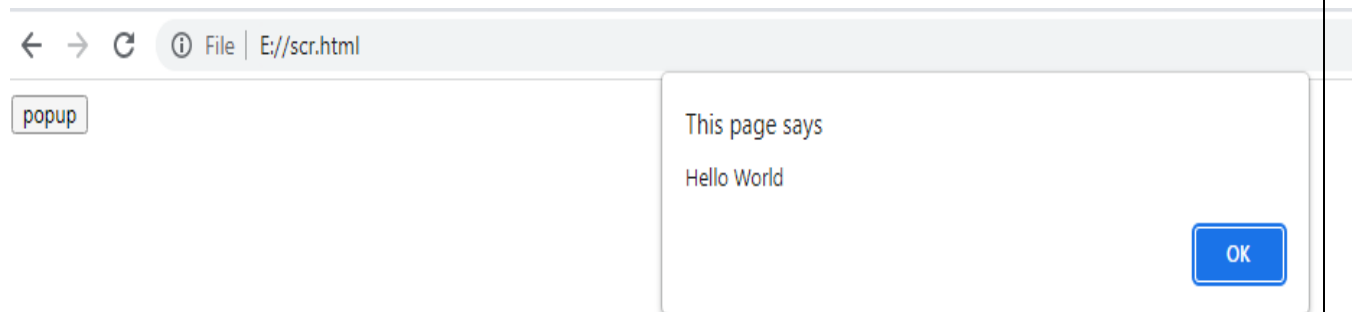3. In an **external file**

For Eg. :

```
<html>
<head>
<script type="text/javascript">
<!--
function popup() {
```

alert("Hello World")

}

//-->

</script>

</head>

<body>

    <input type="button" onclick="popup()" value="popup">

</body>

</html>

Importing an External JavascriptFile :

       Importing an external file is relatively painless. First the file you are importing must be valid Javascript, and only Javascript.

       Second, the file must have the extension "**.js**".

For eg. :

<html>

<head>

    <script **src="myjs.js"**>

    </script>

</head>

<body>

  <input type="button" **onclick="popup()"** value="Click Me!">

</body>

</html>

myjs.js

console.log("Hi Geeks, Welcome to ITE");

Output

Hi Geeks, Welcome to ITE

Some Basic rules for java script syntax:

✓ JavaScript is case sensitive. It is common to start the name of a constructor with a capitalised letter, and the name of a function or variable with a lower-case letter.

✓ Variable declarations: - Not required - Data type not specified

✓ Semi-colons are usually not required, but always allowed at statement end

✓ Arithmetic operators same as Java/C++

✓ Arguments can be any expressions.

✓ Argument lists are comma-separated

✓ Most relational operators syntactically same as Java/C++

Example basic program

```html
<!DOCTYPE html>
<html>

<head>
<title>JavaScript Internal Script</title>
<basehref="https://www.tutorialspoint.com/"/>

<scripttype="text/JavaScript">
functionHello(){
alert("Hello, World");
}
</script>
</head>

<body>
<inputtype="button"onclick="Hello();"name="ok"value="Click Me"/>
</body>

</html>
```

Example 2 by using external java script

function Hello() {

```
alert("Hello, World");

}
```

```
<!DOCTYPE html>
<html>


<head>
<title>Javascript External Script</title>
<scriptsrc="/html/script.js"type="text/javascript"/></script>
</head>


<body>
<inputtype="button"onclick="Hello();"name="ok"value="Click Me"/>
</body>


</html>
```

Form Validation

```
<html>
 <head>
  <title>Title of the document</title>
 <body>
  <h2 style="text-align: center"> Registration Form </h2>
  <form        name="RegForm"        action="/form/submit.php"        onsubmit="return
ValidationForm()" method="post ">
  Name:<input type="text" id="Name" size="60" name="Name">    <br>
E-mail Address:      <input type="text" id="E-mail" size="60" name="Email"> <br>
Password:<input  type="text"  id="Password"  size="60"  name="Password">         <br>
Telephone: <input type="text" id="Telephone" size="60" name="Telephone">     <br>
Select Book <select type="text" value="" name="Subject">
     <option>Python</option>
     <option>HTML</option>
     <option>CSS</option>
```

```html
      <option>PHP</option>
      <option>JavaScript</option>
    </select>
  <br>
    <input type="submit" value="Send" name="Submit">
    <input type="reset" value=Reset" name="Reset">
     </form>
<script>
 function ValidationForm() {
   let username = document.forms["RegForm"]["Name"];
   let email = document.forms["RegForm"]["Email"];
   let phoneNumber = document.forms["RegForm"]["Telephone"];
   let select = document.forms["RegForm"]["Subject"];
   let pass = document.forms["RegForm"]["Password"];
   if(username.value == "") {
    alert("Please enter your name.");
    username.focus();
    return false;
   }
   if(email.value == "") {
    alert("Please enter a valid e-mail address.");
    email.focus();
    return false;
   }
   if(email.value.indexOf("@", 0) < 0) {
    alert("Please enter a valid e-mail address.");
    email.focus();
    return false;        }
   if(email.value.indexOf(".", 0) < 0) {
    alert("Please enter a valid e-mail address.");
    email.focus();
    return false;
   }
   if(phoneNumber.value == "") {
```
19

```
      alert("Please enter your telephone number.");

      phoneNumber.focus();

      return false;

      }

    if(pass.value == "") {

      alert("Please enter your password");

      pass.focus();

      return false;

      }

    if(select.selectedIndex < 1) {

      alert("Please enter your course.");

      select.focus();

      return false;

      }

    return true;

     }

  </script>

 </body>

</html>
```

Registration Form

Name: [   ]

E-mail Address: [   ]

Password: [   ]

Telephone: [   ]

Select Book [▾]

[Send] [Reset]

www.w3docs.com says

Please enter your name.

[OK]